|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Id** | UC-1.1.10 | | |
| **Use Case Name** | Unarchive Notice | | |
| **Created By** | Team Neeraj Soni | **Date Created** | 17/11/2015 |
| **Last Updated By** | Team Neeraj Soni | **Last Revision Date** | 07/12/2015 |
| **Actors** | User | | |
| **Description** | [Provide a brief description of the reason for and outcome of this use case.]  Admin can Unarchive Notice from UI. | | |
| **Trigger** | [Identify the event that initiates the use case. This could be an external business event or system event that causes the use case to begin, or it could be the first step in the normal flow.]  User will click on ‘Un-archive’ link. | | |
| **Preconditions** | [List any activities that must take place, or any conditions that must be true, before the use case can be started. Number each pre-condition.]  1. User must be Logged in as Admin.  2. Archive Notice list must be opened. | | |
| **Postconditions** | [Describe the state of the system at the conclusion of the use case execution. Should include both minimal guarantees (what must happen even if the actor’s goal is not achieved) and the success guarantees (what happens when the actor’s goal is achieved. Number each post-condition.]  1. Notice will be un-archived successfully.  2. User will be remain on same page.  3. Archive Notice list will get updated. | | |
| **Normal Flow** | [Provide a detailed description of the user actions and system responses that will take place during execution of the use case under normal, expected conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description.]  1. User will click on ‘Archive Notice’ Button.  2. List of archive notices from his/her company will be shown.  3. User will click on ‘Un-archive’ Button, corresponding to a particular notice.  4. On successful unarchiving of notice, user will remain on same page.  5. Archive Notice List will get updated. | | |
| **Alternative Flows** | [Document legitimate branches from the main flow to handle special conditions (also known as extensions). For each alternative flow reference the branching step number of the normal flow and the condition which must be true in order for this extension to be executed.] | | |
| **Exceptions** | [Describe any anticipated error conditions that could occur during execution of the use case, and define how the system is to respond to those conditions.]  3a.a If notice is already un-archived by another user (In case of concurrency)   1. Notice not found exception will be thrown. | | |
| **Includes** | [List any other use cases that are included (“called”) by this use case. Common functionality that appears in multiple use cases can be split out into a separate use case that is included by the ones that need that common functionality.] | | |
| **Frequency of Use** | [How often will this Use Case be executed. This information is primarily useful for designers. e.g. enter values such as 50 per hour, 200 per day, once a week, once a year, on demand etc.] | | |
| **Special Requirements** | [Identify any additional requirements, such as nonfunctional requirements, for the use case that may need to be addressed during design or implementation. These may include performance requirements or other quality attributes.] | | |
| **Assumptions** | [List any assumptions that were made in the analysis that led to accepting this use case into the product description and writing the use case description.] | | |
| **Notes and Issues** | [List any additional comments about this use case or any remaining open issues or TBDs (To Be Determined) that must be resolved.] | | |